Encapsulation Considerations

Update at IETF93
draft-ietf-rtgwg-dt-encap-00.txt

Design team

Albert Tian
Erik Nordmark
Jesse Gross
Jon Hudson
Larry Kreeger
Pankaj Garg
Pat Thaler
Tom Herbert

Charter http://www.ietf.org/mail-archive/web/rtgwg/current/msg04715.html

Agenda

- Changes since draft-rtg-dt-encap-01
- Planned changes
- Some outstanding comments from the RTGDIR review
- Larger open issues
  - Scope discussion?
  - More on MPLS?
  - Stronger or more direct recommendations?

Changes #1

- Setting the context that not all common issues might apply to all encapsulations, but that they should all be understood before being dismissed.
- Clarified that IPv6 flow label is useful as entropy together with a UDP source port.
- Editorially added a "summary" set of bullets to most sections.

Changes #2

- Editorial clarifications in the next protocol section to more clearly state the three areas.
- Folded two next protocol sections into one.
- Mention the MPLS first nibble issue in the next protocol section.
- Mention that viewing the next protocol as an index to a table with processing instructions can provide additional flexibility

Changes #3

- For the OAM "don't forward to end stations" added that defining a bit seems better than using a special next-protocol value.
- Added mention of DTLS in addition to IPsec for security.

Changes #4

- Added some mention of IPv6 hop-by-hop options of other headers that potentially can be copied from inner to outer header.
- Added text on architectural considerations when it might make sense to define an additional header/protocol as opposed to using the extensibility mechanism in the existing encapsulation protocol.

Changes #5

- Clarified the "unconstrained TLVs" in the hardware friendly section.
- Clarified the text around checksum verification and full vs. header checksums.
- Added wording that the considerations might apply for encaps outside of the routing area.
- Added more references and removed some

Planned changes

- Fix minor typo in definition of MUST:

  “The capitalized keyword MUST is used as defined in http://en.wikipedia.org/wiki/Julmust”

Thanks to Jamal for finding that typo

RTGDIR review

- Many comments applied in -00
- Need for more explanation for “index to a table containing processing instructions”
- More concrete guidelines please
  - Do the summaries in -00 help?
- Any reference on quiet period when changing the hash?

Scope Discussion

“The degree to which these common issues apply to a particular encapsulation can differ based on the intended purpose of the encapsulation. But it is useful to understand all of them before determining which ones apply.”

- Transport vs. encaps?
- SFC being different than NVO3?

MPLS discussion

- What is already present
  - Entropy label
  - First nibble issue
  - Consider “index to a table with processing instructions” in next protocol context
- What else the WG want to add that is useful across BIER, SFC, NVO3?

Stronger or more direct recommendations/guidelines?

- Please review the summary bullets at the end of each section.
- See end of this slide set in case you don’t want to re-read the whole document!

Next Steps?

Comments please!

Entropy Summary

- The entropy is associated with the transport, that is an outer IP header or MPLS.
- In the case of IP transport use 14 or 16 bits of UDP source port, plus outer IPv6 flowid for entropy.

Next-protocol (non-)summary

- Would it be useful for the IETF come up with a common scheme for encapsulation protocols? If not each encapsulation can define its own scheme.

MTU and Fragmentation Summary

- In some deployments an encapsulation can assume well-managed MTU hence no need for fragmentation and reassembly related to the encapsulation.
- Even so, it makes sense for ingress to track any ICMP packet too big addressed to ingress to be able to log any MTU misconfigurations.
- Should an encapsulation protocol be deployed outside of the original context it might very well need support for fragmentation and reassembly.

OAM Dataplane Support Summary

- It makes sense to reserve a bit for "drop after decapsulation" for OAM out-of-band.
- An encaps needs sufficient extensibility for OAM (such as bits, timestamps, sequence numbers). Might be motivated by in-band OAM but it would make sense to leverage the same extensions for out-of band OAM.
- OAM places some constraints on use of entropy in forwarding devices.
- Should IETF look into error reporting that is independent of the specific encapsulation?

Security Summary

- Need extensibility to be able to add security features (cookies, secure hashes) protecting the encaps header.
- NVO3 likely higher requirements related to isolation, which is in scope for the NVO3 WG.
- Our collective IETF experience is that successful protocols get deployed outside of the original intended context, hence the initial assumptions about the threat model might become invalid. That needs to be considered in the standardization of new encapsulations.

QoS Summary

- Leverage the existing approach in RFC2983 for DSCP handling.

Congestion Summary

- Leverage the existing approach in RFC6040 for ECN handling.
- If the encapsulation can carry non-IP, hence non-congestion controlled traffic, then leverage the approach in draft-ietf-mpls-in-udp
- "Watch this space" for circuit breakers.

Header Protection Summary

- Encapsulations need extensibility to be able to add checksum/CRC for the encapsulation header itself.
- When the encapsulation has a checksum/CRC, include the IPv6 pseudo-header in it.
- The checksum/CRC can potentially be avoided when cryptographic protection is applied to the encapsulation.

Extensibility Summary

- Encapsulations need the ability to be extended to handle e.g., the OAM or security aspects discussed in this document.
- Practical experience seems to tell us that extensibility mechanisms which are not in use on day one might result in immediate ossification by lack of implementation support. In some cases that has occurred in routers and in other cases in middleboxes. Hence devising ways where the extensibility mechanisms are in use seems important.

Hardware Friendly Summary

- Keep the encap header small.
- Put important information at the beginning of the encapsulation header.
- Avoid full packet checksums in the encapsulation if possible. Encapsulations should instead consider adding their own checksum which covers the encapsulation header and any IPv6 pseudo-header.
- Place important information at fixed offset in the encapsulation header.
- Limit the number of header combinations.

NIC Offload Summary

- The considerations for using full UDP checksums are different for NIC offload than for implementations in forwarding devices.
- Be judicious about encapsulations that change fields on a per-packet basis - makes it hard to use TSO.

Middlebox (non-)Summary

- We are likely to see middleboxes that at least parse the headers for successful new encapsulations.
- Should the IETF document considerations for what not to do in such middleboxes?